AMENDMENTS TO THE SPECIFICATION

Please replace the paragraph beginning on page 19, line 22 with the following amended paragraph:

Also included in computer system ~~110~~ 100 of Figure 5 is an alphanumeric input device 106 which in one implementation is a handwriting recognition pad ("digitizer") having regions 106a and 106b (Figure 2A), for instance. Device 106 can communicate information(spatial data and pressure data) and command selections to the central processor 101. System ~~110~~ 100 also includes an optional cursor control or directing device 107 coupled to the bus for communicating user input information and command selections to the central processor 101. In one implementation, device 107 is a touch screen device incorporated with screen 105. device 107 is capable of registering a position on the screen 105 where the stylus makes contact and the pressure of the contact. The display device 105 utilized with the computer system ~~110~~ 100 may be a liquid crystal device, cathode ray tube (CRT), field emission device (FED, also called flat panel CRT) or other display device suitable for creating graphic images and alphanumeric characters recognizable to the user. In one embodiment, the display 105 utilizes color sequential scanning but could also utilize color filters with subpixels.

Please replace the paragraph beginning on page 19, line 22 with the following amended paragraph:

The shortcomings of the example in Figure 10 are overcome in the present invention by using the innovative context packet 520. Figure 11 shows an exemplary task switching diagram in accordance with an embodiment of the present invention, showing five computer programs, three buttons, an interrupt event, and eight context packets. In this example, if the user selects button A 601, program A 603a is invoked and begins running; and button B 605 invokes program B 607a. Programs A ~~601a~~ 603a and B 607a are different from the corresponding programs in Figure 10 in that they have been modified to utilize context packets, in accordance with the present invention. If the user is running program A 603a, he can cause a task switch to program B 607a, and the figure illustrates this with diverging arrows 631, 633, one arrow 631 illustrating the task switch to program B607a, and the other arrow 633 illustrating the creation of context packet A 635 and its storage into memory. As in accordance with the present invention, program A 603a does the following: (1) creates a context packet storing information needed to recreate its state; and (2) stops running and relinquishes any memory it has been temporarily using. While running program A 603a, the user initiates the task switch through the use of a jump function. Continuing with the example, the user can task switch from program B 607a to program C 611a, and then from program C 611a, task switch to program D 613a. At each task swap, a corresponding context packet 637, 639 is created and stored into memory. When running program D 613a, the user can return to program C 611a at the place where he left off. To do this, the user utilizes a return function, which causes: (1) program D 613a to terminate; and (2) program C 611a to be started with context packet C 639 as an input. Upon recognizing context packet C 639 as an input, program C uses the information stored in the context packet 639 to recreate the state that existed in the program 611a when the user's task swap invoked program D 613. The context packets 635, 637, 639, 641 contain enough information for the corresponding program to restore its state when it was last running. Each context packet contains information that is pertinent to the program that created it.

Please replace the paragraph beginning on page 24, line 1 with the following amended paragraph:

Figure 7 is a display screen 105a showing an exemplary email application, having a return button 510 in accordance with an embodiment of the present invention. Other buttons shown are typical of an email application. In this example, n email message, "EMAIL #23", has been received, requesting a meeting. If the user is done with the email program, he can select the return button 510, which invokes a return function, thereby taking him back to the previous program. However, the user may decide to reply to this email, but needs to consult his calendar. The use selects the jump button 512, and this causes a pull-down menu 800 to appear on display screen 150b, as shown in Figure 8. In this example, possible jump programs are: (1) calendar 802; (2) address book 804; (3) to-do list 806; and (4) patient care 808. These four particular possible jump programs are shown because it is useful for the user to task switch to these programs from the email program. Jump program choices can be coded into a program and/or selected by the user. In this example, the first three would be typical jump programs 802, 804, and 806 coded into an email program, but the last one 808 is a user-added choice to manage the care of patients, which is an application designed for physicians.